REMARKS

Claims 5, 10, 15, and 16 are amended.

Claims 1-16 are pending.

Regarding Proposed Changes to the Drawings

In proposed **Fig. 1**, reference number "220" was changed to -- 200 -- to properly match the description beginning on page 4, line 22.

In proposed **Fig. 3**, the overall method's reference number was changed from "402" to -- 400 -- to properly match the description on page 9, line 12.

Consequently, it is respectfully requested that the objections to the drawings under 35 U.S.C. §1.83(a) be reconsidered and withdrawn.

Regarding Claim Amendments

Claims 5, 10, 15, and 16 are amended herein to provide proper grammatical/antecedent language to selected elements. These amendments were not made based on the cited art and/or for any substantive purpose relating to patentability.

Rejections under 35. U.S.C. §102(b)

Claims 1-16 stand rejected under 35 U.S.C. §103(b) as being anticipated by an article titled "Hardware Design of a 2-D Motion Estimation System Based on the Hough Transform", by *Hsiang-Ling Li et al.* and published in IEEE Transactions on Circuits and Systems – II: Analog and Digital Signal Processing, Vol. 45, No. 1, pages 80-95 in January 1998 (IEEE Article No. 1057-7130/98).

6

1   The Applicant respectfully traverses each of these rejections for at least the

2 following reasons.

3   As originally stated in the pending application:

4 The Hough Transform is a well-known computer vision algorithm that can
be implemented to robustly detect a wide variety of features such as lines,

5 circles, and anything else that can be readily parameterized or otherwise

6 cast in terms of a discrete popularity algorithm. Unfortunately, the Hough
transform tends to be computationally intensive. Many personal computer

7 (PC) microprocessors are unable to provide the requisite processing

8 capability usually associated with the Hough Transform. (page 1, line 21)

9 One of the problems with past implementations of the Hough transform is
that the arrays of accumulators needed for Hough transforms have

10 historically needed to be low-resolution because of the data storage expense

11 and execution time requirements for higher resolution arrays. This is
because, high-resolution arrays require both more time to manage, and more

12 time to analyze after the algorithm has finished processing the
accumulators. As described below, the various methods and arrangements

13 describe ways to leverage dedicated hardware to solve these problems. As
will be shown, it is feasible to use an inexpensive processor that can operate

14 on ordered data to implement the Hough transform. (page 4, line 2)

15

16   Hence, the specification clearly states that various systems exist which

17 implement the Hough transform. However, the logic required to perform the

18 Hough transform can be complex and/or expensive. That is why the industry is

19 busy working on improved systems that use different solutions, optimizations,

20 approximations, etc.

21   With this in mind, *Hsiang-Ling Li et al.* disclose a motion estimation system

22 that implements a straight-line Hough transform through custom designed logic

23 based on a feature-domain algorithm. *Hsiang-Ling Li et al.* disclose that their goal

24 is to develop a less computationally intensive algorithm (page, 80, second column,

25 first paragraph).

7

G:\MSI-0\66J US\MSI-661US.MOI.doc

Hsiang-Ling Li et al. disclose an improved algorithm that is implemented in custom logic that includes a preprocessor block (see, Fig. 4) and a Hough Transform (HT) processor block (see, e.g., Fig. 5). The preprocessor block performs boundary-smoothing, contour tracing, contour integrating and dominant point detection. The HT processor block then processes contour segments based on the dominant points and using a vote accumulator estimates a translation vector. See, e.g., sections III-V, beginning on page 82.

Hsiang-Ling Li et al. implemented their preprocessor and HT processor blocks using conventional gate logic simulation tools available from Mentor Graphics® (see, e.g., Abstract, page 80). Thus, *Hsiang-Ling Li et al.* essentially disclose an improved algorithm that is taught and implemented using specially designed logic blocks. These specially designed logic blocks may be implemented through custom VLSI logic, standard cell logic, and/or FPGA logic (page 80, second column).

To the contrary, Claims 1-16 are directed towards implementing a Hough transform using a host processor and a graphics processor. Hence, there is no need to design and/or build a special purpose logic circuit as taught by *Hsiang-Ling Li et al.* One need only configure (e.g., program) a host processor and graphics preprocessor accordingly.

Hsiang-Ling Li et al. clearly fail to disclose or otherwise even suggest that a Hough transform may be efficiently performed by splitting up the processing between a host processor and a graphics processor as recited in the pending claims. Moreover, the algorithm taught by *Hsiang-Ling Li et al.* is different that being claimed.

8

The Office Action suggests that the "Mentor Graphics tools inherently provide a graphics processor". This statement is inherently wrong. As well known in the art, Mentor Graphic's tools are part of an electronic design automation (EDA) software product that allows engineers to develop logic circuits, layout the logic circuits, simulate the operation of the logic circuits, and output resulting logic design information suitable for manufacturing corresponding hardware/firmware. Thus, Mentor Graphic's tools are actually software applications for use in a computer. The computer instructions that form this software application are therefore processed as would any other application's instructions via the CPU of the computer.

Attention is now drawn to the pending claims.

Independent **Claim 1** is directed towards a method that includes providing image data and performing a Hough transform on the image data using a host processor and an operatively configured graphics processor. For at least the reasons stated above, this method is not disclosed or suggested by *Hsiang-Ling Li et al.*

Claim 2, which depends from Claim 1, further specifies that the graphics processor is configured to count votes in a resulting Hough transform voting buffer. *Hsiang-Ling Li et al.* also fail to disclose or suggest the use of a graphics processor to count votes.

Claim 3, which depends from Claim 1, further specifies that the graphics processor is configured to convolve image values and provide corresponding results to the host processor. Again, *Hsiang-Ling Li et al.* fail to disclose or suggest the use of a graphics processor in this manner.

9

Claim 4, which also depends from Claim 1, further specifies that the graphics processor performs an alpha-blending operation that selectively increments accumulators that correspond to parameter combinations that are likely associated with an observation. *Hsiang-Ling Li et al.* do not disclose or suggest the use the alpha-blending operation of a graphics processor to perform such a method.

In **Claim 5**, which depends from Claim 1, it is further recited that the graphics processor performs a histogram computation to find the maxima value in a Hough transform voting buffer. The logic taught by *Hsiang-Ling Li et al.*, which does not even include a graphics processor, does not perform a histogram computation to find a maxima value in a Hough transform voting buffer.

Independent **Claim 6** is directed towards an apparatus that includes a host processor and a graphics processor. As recited the host processor is configured to provide image data and the graphics processor is configured to perform selected steps of a Hough transform algorithm on the image data in association with the host processor. For at least the reasons stated above, this apparatus is not disclosed or suggested by *Hsiang-Ling Li et al.* Instead, *Hsiang-Ling Li et al.* teach that special purpose hardware or firmware (e.g., VLSI, FPGA) logic can be used to perform a straight-line Hough transform.

Claim 7, which depends from Claim 6, the apparatus further includes local memory that is operatively coupled to the graphics processor. Here, the graphics processor is configured to count votes in a resulting Hough transform voting buffer within the local memory. *Hsiang-Ling Li et al.* fail to disclose or suggest this apparatus.

10

G:\WSI-0\661US\WSI-661US.M01.doc

Claim 8, which depends from Claim 6, further specifies that the graphics processor is configured to convolve image values and provide corresponding results to the host processor. Again, *Hsiang-Ling Li et al.* fail to disclose or suggest the use of a graphics processor in this manner.

Claim 9, which also depends from Claim 6, further specifies that the apparatus includes local memory and that the graphics processor performs an alpha-blending operation that selectively increments accumulators within the local memory that correspond to parameter combinations that are likely associated with an observation. *Hsiang-Ling Li et al.* do not disclose or suggest using the alpha-blending operation of a graphics processor in this way.

Claim 10, which depends from Claim 6, further recites that the apparatus includes a local memory and that the graphics processor performs a histogram computation to find the maxima value in the Hough transform voting buffer within the local memory. The logic taught by *Hsiang-Ling Li et al.*, which does not even include a graphics processor, does not perform a histogram computation to find a maxima value in a Hough transform voting buffer.

Independent **Claim 11** is directed towards a computer-readable medium having computer-executable instructions for performing steps that include providing image data and performing a Hough transform on the image data using a host processor and an operatively configured graphics processor. For at least the reasons stated above, this computer-readable medium is not disclosed or suggested by *Hsiang-Ling Li et al.*

Claim 12, which depends from Claim 11, recites that the computer-executable instructions cause the graphics processor to count votes in a resulting

11

G:\MSI-0661US\MSI-001US.M01.doc

PAGE 13/18 * RCVD AT 1/21/2004 6:21:23 PM [Eastern Standard Time] * SVR:USPTO-EFXRF-1/3 * DNIS:8729306 * CSID:3035390271 * DURATION (mm-ss):04-50

Hough transform voting buffer. *Hsiang-Ling Li et al.* fail to disclose or suggest this.

Claim 13, which depends from Claim 11, recites that the computer-executable instructions cause the graphics processor is to convolve image values and provide corresponding results to the host processor. Again, *Hsiang-Ling Li et al.* fail to disclose or suggest the use of a graphics processor in this manner.

Claim 14, which also depends from Claim 11, further states that the computer-executable instructions cause the graphics processor to perform an alpha-blending operation that selectively increments accumulators that correspond to parameter combinations that are likely associated with an observation. *Hsiang-Ling Li et al.* do not disclose or suggest using the alpha-blending operation of a graphics processor in this way.

In **Claim 15**, which depends from Claim 11, the computer-executable instructions further cause the graphics processor to perform a histogram computation to find a maxima value in a Hough transform voting buffer. *Hsiang-Ling Li et al.* do not even use a graphics processor and do not perform a histogram computation to find a maxima value in a Hough transform voting buffer.

A method is recited in independent **Claim 16**. The method includes causing dedicated graphics hardware to support at least one step associated with a Hough transform algorithm. Thus, the dedicated graphics hardware may quantize a bounded portion of a parameter space that may contain a desired feature. *Hsiang-Ling Li et al.* do not disclose or suggest this. The dedicated graphics hardware may, for each discrete quantized parameter combination, allocate an incrementable accumulator. *Hsiang-Ling Li et al.* do not disclose or suggest having a graphics processor do this. The dedicated graphics hardware may gather observations that

12

1   can be mapped into the parameter space. This is not taught by *Hsiang-Ling Li et*

2   *al.* The dedicated graphics hardware may, for each observation, increment each

3   of the accumulators that corresponds to parameter combinations that may have

4   produced the observation. *Hsiang-Ling Li et al.* do not disclose or suggest this.

5   The dedicated graphics hardware may determine a maxima in a resulting quantized

6   parameter array and the corresponding parameter combinations. This also is not

7   taught by *Hsiang-Ling Li et al.*

8   For at least the reasons stated above, it is, therefore, respectfully requested

9   that the rejections be reconsidered and withdrawn.

10

11   **Conclusion**

12   The pending claims have been placed in condition for allowance and are

13   clearly patentable over the cited art and should therefore be allowed.

14

15   Respectfully Submitted,

16

17   Date: _1/21/2004_   By: _____

18   Thomas A. Jolly
                                         Reg. No. 39,241

19

20

21

22

23

24

25

13